State of Montana Project Management Office

Project Execution and Approval Phase

Integration Test Plan Instructions

Outline for a plan for integration testing, in this case integration containing hardware and software. This outline is easily adaptable to a software-only integration plan. Every section of the outline is annotated to describe typical contents of the section.

Helps the technical team think through the logical sequence of integration activities so that their individual detailed development plans are well synchronized and integration happens in a reasonable (not helter-skelter) manner. Helps the project manager plan for equipment and lab availability. Elements such as Completion Criteria help ensure that the test results are reviewed and understood and a quality product is handed off to system-level testing. (Highly buggy stuff given to the system test group is a sure way to delay the project.)

1. Create a draft of the plan very early in the Execution Phase. By this time you'll know from the high-level design work in the Initiation phase what major pieces of hardware and/or software will have to be integrated. An early look at integration sequencing will help you sanity check the amount of time you've put into the schedule for integration testing.

2. Review the plan with technical team members and others who will be supporting integration testing.

3. The plan can be refined as appropriate as the detailed designs are completed.

4. The outcome of reviews should be a plan that everyone agrees will thoroughly test the "bringing together" of the system, before it is handed off for system or functional testing. The goal is to ensure no major interface issues remain undiscovered until system testing. If that were to happen, the schedule could be severely impacted while those basic interface problems are corrected by developers.

5. Ensure that any equipment or lab needs for integration testing are communicated to anyone who will be responsible for getting those resources in place on time.

Integration Test Plan Outline

(Annotated with Typical Contents)

Introduction

1. Revision History. Record all revisions to the document.

2. Purpose and Scope. State the purpose and scope of the document.

3. List of Reference Documents

   List all reference documents; the minimum are those listed here plus documentation on tools and test equipment, integration station, etc. For instance:

   - Product Architecture Document
   - High Level Software Design
   - High Level Hardware Design
   - Hazard Analysis

4. Integration Strategy

5. Entry Criteria.

   Specify the criteria that must be met before integration of specific elements may begin (e.g., functions must have been unit tested).

6. Elements to be Integrated.

   Identify all the elements to be integrated by subsystem into which they will be integrated.

7. Integration Strategy.

   Describe the integration approach (top-down, bottom-up, functional groupings, etc.) and the rationale for choosing that approach.

8. Sequence of Feature/Function and Hardware/Software Integration.

NOTE: *The sections below are one suggested approach. Organize this section to best describe the integration strategy identified in above. But keep in mind the key goal of tying integration activities, especially software builds, to identifiable feature/functionality.*

9. Software Integration Sequence. For each subsystem:

   Identify the sequence in which the software code functions and modules will be integrated. Relate this sequence to any product features/functions that are being built up. Specify any hardware dependencies for early software integration activities.

10. Hardware Integration Sequence. For each subsystem:

Identify the sequence in which the hardware elements will be integrated. Specify any software dependencies for early hardware integration activities.

11. Full Hardware/Software Integration Sequence. For each subsystem:

    Identify the sequence of integrating software build(s) with the hardware modules. Identify the product features/functions that are being built up at each step.

12. Subsystem Integration Sequence. Identify the order in which subsystems will be integrated.

13. Individual Steps and Test Description.

    For each step of the integration process identified above, describe the type of tests that will be used to verify that the elements integrated in this step perform as expected. Describe in general the expected results of the test set. (NOTE: This is not a detailed description of test protocols. Think of this as the test design phase. Specific protocols will be written to fulfill the goals of the tests identified in this section.)

    At the lower levels, these tests will focus on direct testing of interfaces between software functions and modules, between software functions or modules and specific electronics, etc. As more of the system is put together, tests will continue to be concerned with verifying interfaces, but will include more of a "functional test" flavor.

    At each step, specify where re-running of tests from previous steps may be required to verify that integrating this new element has not caused previously tested functions to fail.

14. Software Integration Test Description.

15. Hardware Integration Test Description.

16. Hardware/Software Integration Test Description.

17. Subsystem Integration Test Description.

18. Final Functional Tests.

    Describe the set of functional tests to be run at the end of integration to verify the basic functionality of the system. These tests are intended to confirm that the system has been successfully integrated and is now ready for full System Test by Engineering. NOTE: The desired tests may have already been defined as part of the functional test set created for System Test. Consult the System Test Plan and protocols for any tests that can be borrowed.

19. Tools and Test Equipment Required.

Identify all tools and test equipment needed to accomplish the integration. Examples are computer workstations, oscilloscopes, meters, host operating systems, etc. Specify revision levels or version numbers where necessary.

The integration protocols will specify in detail all configuration information, set-up procedures, parameters, environmental conditions (lighting, humidity, etc.), etc., necessary for the integration testing. In this plan, summarize any significant configuration efforts, special environmental requirements, etc.

20. Program Stubs and Test Data Required.

    Based on the testing strategy and test design, identify any program stubs or special test data required for each integration step.

21. Responsibilities and Schedule.

    Identify all personnel skill types and quantities, define the responsibilities assigned to each and develop a schedule which shows the sequence of major test set execution and the amount of time estimated for that test set. Identify known risks and any assumptions. NOTE: If any of this information is included in the project plan instead, reference the section in that document.

22. Roles and Responsibilities.

23. Key Dependencies - Define key dependencies in the schedule (people or equipment resources, availability of other hardware or software under development, etc.).

24. Risks and Assumptions.

25. Schedule.

26. Problem Recording and Resolution.

    Define the mechanism to be used for problem recording and resolution, including any necessary rework of documents, software or hardware elements, test plan or procedures. Include (or reference) any forms. Reference any standard problem tracking procedures that will be used.

27. Rework, Review and Retest Procedures.

    Define the process for rework, review and retest of any element needing modification. This process must include sufficient retest to verify that any modifications have not impacted other functions already tested. This process must also define the project guidelines for how related design and requirements documents will be updated as changes are made. *Change management processes for this phase must be defined and followed.*

28. Suspension, Restart and Exit Criteria.

Define criteria for a) suspending testing before completion (for instance, the discovery of major problems with a certain feature; b) criteria for restarting testing following such a suspension; and c) the criteria for determining that integration test is complete and system test can begin.

Administrative Information

Revision	Author	Date	Sections Affected	Change Summary
1.0		1/3/2009		

Current Version	1.0
Date	1/3/2009
Master Document Chapter Number	5
Document ID	110